

ALTAS
MANUAL DE USUARIO
SERVICIO PTFR

Versión 1.1

Unidad de Arquitectura del Software



Agencia de Informática y Comunicaciones de la Comunidad de Madrid



Hoja de Control

Título	Manual de Usuario Servicio de PTFR		
Documento de Referencia	NORMATIVA ATLAS		
Responsable	Unidad de Arquitectura del Software		
Versión	1.1	**Fecha Versión**	04/11/2014

Registro de Cambios

Versión	Causa del Cambio	Responsable del Cambio	Fecha del Cambio
1.0	Versión inicial del documento	Unidad de Arquitectura del Software	09/06/2014
1.1	Documentados métodos especiales	Unidad de Arquitectura del Software	04/11/2014

Índice

1. INTRODUCCIÓN

Este documento contiene instrucciones y ejemplos de uso de la librería de servicios de PTFR. Para más información sobre las capacidades funcionales de este servicio, debe consultarse la documentación original del proyecto.

La librería se compone del siguiente servicio:

- **PtfrService** – Servicios de administración electrónica para el intercambio de datos con Portafirmas

1.1. Audiencia objetivo

El lector objetivo de este documento es toda aquella persona que esté desarrollando una solución en la que se necesite usar servicios de PTFR en sus aplicación ATLAS.

1.2. Conocimientos Previos

Para un completo entendimiento del documento, el lector deberá tener conocimientos previos sobre las siguientes tecnologías:

- Lenguaje Java
- Spring Framework

2. DESCRIPCIÓN

El servicio de Spring “**PtfrService**” que incluye el módulo de administración electrónica de Atlas, provee las siguientes funcionalidades, que se explican en los siguientes apartados:

- consultaEstadoDoc
- solicitarDocumentos
- consultaFirmantesTipoDocumento
- consultaTiposDocumento
- consultaFirmantes
- borrarDocumentosFinalizados
- eliminarDocumento
- altaSolicitud
- altaSolicitudConAnexos
- consultaEtapasDoc
- Enviar
- EnviarOffline

3. INSTALACIÓN Y CONFIGURACIÓN

3.1. INSTALACIÓN

El Servicio '***PtfrService*** está incluido en la librería ***atlasfrm-eadm-lib***. Para poder utilizarlo, bastará con incluir la dependencia de este módulo en el fichero *pom.xml* del proyecto, añadir la ruta correspondiente al fichero *xml* de Spring en la variable '*configLocations'*, y definir en el contexto de Spring la configuración del servicio para la aplicación., según se describe en los siguientes apartados:

3.1.1. Paso 1: Añadir la dependencia al módulo de Administración Electrónica

Atención
Este paso no es necesario si el proyecto se ha generado a partir del arquetipo *web, servicioweb* o *documentumweb* de Atlas, ya que dicha librería ya viene incluida por defecto en dichos arquetipos.

Para añadir la dependencia al módulo de seguridad dentro de nuestro proyecto *maven* hay que añadir una entrada "**dependency**" en la sección "**dependencies**" del fichero "**pom.xml**" del proyecto, como se puede ver en el siguiente ejemplo:

pom.xml

```
<dependencies>
      [...]
      <dependency>
          <groupId>atlasfrm</groupId>
          <artifactId>atlasfrm-eadm-lib</artifactId>
          <version>${atlasfrm-eadm-lib.version}</version>
      </dependency>
      [...]
</dependencies>
```

3.1.2. Paso 2: Incluir la ruta de definición del servicio ‘PtfrService’

La librería de administración electrónica ya contiene una definición del servicio ‘**PtfrService’** dentro del contexto de Spring. Bastará con incluir la siguiente ruta en una sentencia <import> dentro de un fichero de configuración de Spring (por ejemplo, en el fichero applicationContext-general.xml):

applicationContext-general.xml

```
<import resource="classpath:/conf/applicationContext-eadm-lib.xml"/>
```

3.1.3. Paso 4: Inyectar el Servicio en las clases que lo requieran

Para inyectar el Servicio previamente definido, se pasa como “**property**” al bean que lo vaya a usar, que debe tener el método “**set**” correspondiente. Para más información sobre cómo inyectar referencias, se puede consultar la documentación de Spring.

applicationContext-services.xml

```
<bean id="miService" class="xxxx.services.MiServiceImpl"
      p:eregAltaService-ref="ptfrService"/>
```

xxxx.services.MiServiceImpl.java

```
public class MiServiceImpl implements MiService {

    private PtfrService ptfrService;

    public void setPtfrService(PtfrService ptfrService) {
        this.ptfrService = ptfrService;
    }
    ...
}
```

4. USO DEL SERVICIO PtfrService

Una vez configurado el entorno para el uso del servicio, puede procederse a su utilización.

4.1. Método consultaEstadoDoc

PtfrService.java

```
    /**
     * Consulta el estado de un documento. Únicamente devuelve el estado del
     * documento, formado por una identificador en forma de carácter en mayúscula
     * @param codAplicacion Código de la aplicación
     * @param referencia Refencia del documento a consultar
     * @param codigoArchivo Código del archivo
     * @return {PtfrEstadoDocOut}
     * @throws ServiceException
     */
    PtfrEstadoDocOut consultaEstadoDoc(String codAplicacion, String referencia,
            String codigoArchivo) throws ServiceException;
```

Objeto de salida
Todos los servicios de Ptfr que devuelvan un objeto heredarán de PtfrRespuestaBase

PtfrRespuestaBase.java

```
/**
 * Clase Base de la que heredarán todas las clases OUT de los
 * métodos del servicio
 * codigoRespuesta código de respuesta recibido del WS
 * descripcionRespuesta descripción de la respuesta recibida
 *
 */
public class PtfrRespuestaBase
```

PtfrEstadoDocOut .java

```
/** Respuesta del método consultaEstadoDoc
 * {atlas.eadm.services.PtfrService#consultaEstadoDoc}
 * Hereda de {PtfrRespuestaBase}
 * PtfrDocumento
 *
 */
public class PtfrEstadoDocOut extends PtfrRespuestaBase
```

PtfrDocumento.java

```
/**
 * Objeto para guardar la información de un documento
 * estado Estado del documento
 * codArchivo Código del archivo
 * codArchivoOriginal Código del archivo original
 * referencia Referencia del documento a consultar
 * referenciaSolicitante Referencia del solicitante
 * fechaAccion Fecha de la accion
 * motivorechazo Motivo del rechazo
 */

public class PtfrDocumento
```

4.2. Método solicitarDocumentos

PtfrService.java

```
   /**
     * Retorna información acerca de los documentos asociados a la aplicación
     * especificada. No retorna el contenido de los documentos en sí, sino datos
     * informativos, como el código de archivo, referencia, etc
     *
     * @param codigoAplicacion Código de la aplicación
     * @return {PtfrSolicitarDocumentosOut}
     * @throws ServiceException
     */
    PtfrSolicitarDocumentosOut solicitarDocumentos(String codigoAplicacion)
            throws ServiceException;
```

Objeto de salida
Todos los servicios de Ptfr que devuelvan un objeto heredarán de PtfrRespuestaBase

PtfrEstadoDocOut .java

```
/** Respuesta del método solicitarDocumentos
 * {atlas.eadm.services.PtfrService#solicitarDocumentos}
 * Hereda de {PtfrRespuestaBase}
 * Lista de PtfrDocumento
 *
 */

public class PtfrSolicitarDocumentosOut extends PtfrRespuestaBase
```

PtfrDocumento.java

```java
/**
 * Objeto para guardar la información de un documento
 * estado Estado del documento
 * codArchivo Código del archivo
 * codArchivoOriginal Código del archivo original
 * referencia Referencia del documento a consultar
 * referenciaSolicitante Referencia del solicitante
 * fechaAccion Fecha de la accion
 * motivorechazo Motivo del rechazo
 */

public class PtfrDocumento
```

4.3. Método consultaFirmantesTipoDocumento

PtfrService.java

```java
    /**
     * Obtiene información acerca de los firmantes del documento indicado
     * @param codAplicacion Código de la aplicación
     * @param codDocumento Código del tipo de documento a consultar
     * @return {PtfrConsultaFirmantesTipoDocOut}
     * @throws ServiceException
     */
    public PtfrConsultaFirmantesTipoDocOut consultaFirmantesTipoDocumento(String
codAplicacion, int codDocumento)
        throws ServiceException;
```

Objeto de salida
Todos los servicios de Ptfr que devuelvan un objeto heredarán de PtfrRespuestaBase

PtfrEstadoDocOut .java

```java
/** Respuesta del método consultaFirmantesTipoDocumento
 * {atlas.eadm.services.PtfrService#consultaFirmantesTipoDocumento}
 * Hereda de {PtfrRespuestaBase}
 * Lista de PtfrFirmanteTipoDoc
 *
 */

public class PtfrConsultaFirmantesTipoDocOut extends PtfrRespuestaBase
```

PtfrFirmanteTipoDoc .java

```
/**
 * Clase utilizada para guardar información del tipo
 * documento de un firmante
 * tipo Tipo de documento del firmante
 * cargo Cargo del firmante
 * etapa Número de etapa
 */
public class PtfrFirmanteTipoDoc
```

4.4. Método consultaTiposDocumento

PtfrService.java

```
    /**
     * Obtiene información sobre los tipos de documentos vinculados a una aplicación
     * y módulo concretos
     * @param codAplicacion Código de la aplicación
     * @param codModulo Código del módulo a consultar
     * @return {PtfrConsultaTiposDocumentoOut}
     * @throws ServiceException
     */
    public PtfrConsultaTiposDocumentoOut consultaTiposDocumento(String
codAplicacion, String codModulo)
        throws ServiceException;
```

Objeto de salida
Todos los servicios de Ptfr que devuelvan un objeto heredarán de PtfrRespuestaBase

PtfrEstadoDocOut .java

```
/** Respuesta del método consultaTiposDocumento
 * {atlas.eadm.services.PtfrService#consultaTiposDocumento}
 * Hereda de {PtfrRespuestaBase}
 * Lista de PtfrTipoDocumento
 *
 */

public class PtfrConsultaTiposDocumentoOut extends PtfrRespuestaBase
```

PtfrFirmanteTipoDoc .java

```
/**
 * Clase utilizada para guardar información del tipo
 * documento de un firmante
 * tipo Tipo de documento del firmante
 * cargo Cargo del firmante
 * etapa Número de etapa
 */
public class PtfrFirmanteTipoDoc
```

4.5. Método consultaFirmantes

PtfrService.java

```
    /**
      * Devuelve todos los firmantes asociados a una aplicación o módulo determinado
      * @param codAplicacion Código de la aplicación
      * @param codModulo Código del módulo a consultar
      * @return {PtfrConsultaFirmantesOut}
      * @throws ServiceException
      */
        public PtfrConsultaFirmantesOut consultaFirmantes(String codAplicacion,
String codModulo) throws ServiceException;
```

Objeto de salida
Todos los servicios de Ptfr que devuelvan un objeto heredarán de PtfrRespuestaBase

PtfrConsultaFirmantesOut.java

```
/** Respuesta del método consultaFirmantes
 * {atlas.eadm.services.PtfrService#consultaFirmantes}
 * Hereda de {PtfrRespuestaBase}
 * Lista de PtfrFirmante
 *
 */

public class PtfrConsultaFirmantesOut extends PtfrRespuestaBase {
```

PtfrFirmante.java

```
/**
 * Clase utilizada para guardar información de un firmante
 * codUsu Código del firmante
 * nombre Nombre del firmante
 * cargo Cargo del firmante
 */
public class PtfrFirmante
```

4.6. Método borrarDocumentosFinalizados

PtfrService.java

```
    /**
     * Solicita la eliminación de los documentos finalizados, asociados a la
     * aplicación indicada
     * @param codigoAplicacion Código de la aplicación
     * @return {PtfrBorrarDocumentosFinalizadosOut}
     * @throws ServiceException
     */
    public PtfrBorrarDocumentosFinalizadosOut borrarDocumentosFinalizados(String
codigoAplicacion)
            throws ServiceException;
```

Objeto de salida
Todos los servicios de Ptfr que devuelvan un objeto heredarán de PtfrRespuestaBase

PtfrConsultaFirmantesOut.java

```
/** Respuesta del método borrarDocumentosFinalizados
 * {atlas.eadm.services.PtfrService#borrarDocumentosFinalizados}
 * Hereda de {PtfrRespuestaBase}
 * Lista de PtfrDocumento
 *
 */

public class PtfrBorrarDocumentosFinalizadosOut extends PtfrRespuestaBase
```

PtfrDocumento.java

```
/**
 * Objeto para guardar la información de un documento
 * estado Estado del documento
 * codArchivo Código del archivo
 * codArchivoOriginal Código del archivo original
 * referencia Referencia del documento a consultar
 * referenciaSolicitante Referencia del solicitante
 * fechaAccion Fecha de la accion
 * motivorechazo Motivo del rechazo
 */
public class PtfrDocumento
```

4.7. Método eliminarDocumento

PtfrService.java

```
    /**
     * Solicita la eliminación de un documento concreto.
     * @param codAplicacion Código de la aplicación
     * @param referenciaDocumento Referencia de la solicitud
     * @param codigoArchivoOriginal Código del archivo
     * @return {PtfrEliminarDocumentoOut}
     * @throws ServiceException
     */
    public PtfrEliminarDocumentoOut eliminarDocumento(String codAplicacion, String
referenciaDocumento, String codigoArchivoOriginal)
            throws ServiceException;
```

Objeto de salida
Todos los servicios de Ptfr que devuelvan un objeto heredarán de PtfrRespuestaBase

PtfrEliminarDocumentoOut.java

```
/** Respuesta del método eliminarDocumento
 * {atlas.eadm.services.PtfrService#eliminarDocumento}
 * Hereda de {PtfrRespuestaBase}
 * documento {PtfrDocumento}
 */

public class PtfrEliminarDocumentoOut extends PtfrRespuestaBase;
```

4.8. Método altaSolicitud

PtfrService.java

```
    /**
     * Realiza el alta de una solicitud de documentos.
     * @param datosEntrada datos entrada de tipo {@link PtfrAltaSolicitudIN}
     * @return {PtfrAltaSolicitudOut}
     * @throws ServiceException
     */
    public PtfrAltaSolicitudOut altaSolicitud(PtfrAltaSolicitudIN datosEntrada)
            throws ServiceException;
```

Objeto de salida
Todos los servicios de Ptfr que devuelvan un objeto heredarán de PtfrRespuestaBase

PtfrAltaSolicitudOut.java

```
/** Respuesta del método altaSolicitud
 * {atlas.eadm.services.PtfrService#altaSolicitud}
 * Hereda de {PtfrRespuestaBase}
 * Lista de PtfrDocumento
 *
 */

public class PtfrAltaSolicitudOut extends PtfrRespuestaBase {
```

PtfrDocumento.java

```
/**
 * Objeto para guardar la información de un documento
 * estado Estado del documento
 * codArchivo Código del archivo
 * codArchivoOriginal Código del archivo original
 * referencia Referencia del documento a consultar
 * referenciaSolicitante Referencia del solicitante
 * fechaAccion Fecha de la accion
 * motivorechazo Motivo del rechazo
 */
public class PtfrDocumento
```

Parámetro de entrada

PtfrAltaSolicitudIN.java

```
/**
* Objeto con los datos de entrada para el método altaSolicitud de
* {@link atlas.eadm.services.PtfrService#altaSolicitud(PTFRAtlaSolicitudIN)}
* codigoAplicacion Código de la aplicación
* codigoTipoDocumento Código del tipo de documento
* fichero adjunto con documento
* referenciaDocumento Referencia del documento
* codigoArchivo Código de archivo
* modoRespuesta Modo de la respuesta que hay que dar
* referenciaSolicitante Referencia del solicitante
* titulo título del documento
* cove indica si hay que añadir el código COVE
* lista de PtfrEtapas
*/
public class PtfrAltaSolicitudIN
```

PtfrEtapas.java

```
/**
 * Clase utilizada para guardar información de
 * una etapa
 * codigoFirmante Código del firmante
 * orden Orden de la etapa
 * firmante Código del firmante
 * estado Estado de la etapa
 * fecha Fecha de la firma
 */
public class PtfrEtapas
```

El método incluirFirmante obtiene como parámetro un String con el firmante a añadir. Se puede llamar tantas veces como sea necesario

Método incluirFirmante

```
    /**
     * Añade una etapa a la solicitud
     * @param codigoFirmante String con el código del firmante
     */
public void incluirFirmante(String codigoFirmante)
```

4.9. Método altaSolicitudConAnexos

PtfrService.java

```
        /**
     * Realiza el alta de una solicitud de documentos con anexos.
     * @param datosEntrada datos entrada de tipo {@link
PtfrAltaSolicitudConAnexosIN}
     * @return {PtfrAltaSolicitudConAnexosOut}
     * @throws ServiceException
     */
    public PtfrAltaSolicitudConAnexosOut
altaSolicitudConAnexos(PtfrAltaSolicitudConAnexosIN datosEntrada)
            throws ServiceException;
```

Objeto de salida

Todos los servicios de Ptfr que devuelvan un objeto heredarán de PtfrRespuestaBase

PtfrAltaSolicitudConAnexosOut.java

```
/** Respuesta del método altaSolicitudConAnexos
 * {atlas.eadm.services.PtfrService#altaSolicitudConAnexos}
 * Hereda de {PtfrRespuestaBase}
 * Lista de listaAnexos
 * documento {PtfrDocumento}
 */

public class PtfrAltaSolicitudConAnexosOut extends PtfrRespuestaBase {
```

PtfrDocumento.java

```
/**
 * Objeto para guardar la información de un documento
 * estado Estado del documento
 * codArchivo Código del archivo
 * codArchivoOriginal Código del archivo original
 * referencia Referencia del documento a consultar
 * referenciaSolicitante Referencia del solicitante
 * fechaAccion Fecha de la accion
 * motivorechazo Motivo del rechazo
 */
public class PtfrDocumento
```

PtfrAnexo.java

```
/**
 * Clase utilizada para guardar información
 * de un anexo
 * codigoArchivo Código del archivo
 * codArchivoOrig Código original del archivo
 */
public class PtfrAnexo
```

Parámetro de entrada

PtfrAltaSolicitudConAnexosIN.java

```
/**
* Objeto con los datos de entrada para el método altaSolicitudConAnexos de
* {@link
atlas.eadm.services.PtfrService#altaSolicitudConAnexos(PTFRAtlaSolicitudConAnexosIN)}
* codigoAplicacion Código de la aplicación
* codigoTipoDocumento Código del tipo de documento
* referenciaDocumento Referencia del documento
* codigoArchivo Código de archivo
* modoRespuesta Modo de la respuesta
* referenciaSolicitante Referencia del solicitante
* titulo Titulo del documento
* cove Indica si hay que añadir el código COVE
* Lista de {PtfrAnexo}
* Lista de {PtfrEtapas}
*/
public class PtfrAltaSolicitudConAnexosIN
```

PtfrEtapas.java

```
/**
 * Clase utilizada para guardar información de
 * una etapa
 * codigoFirmante Código del firmante
 * orden Orden de la etapa
 * firmante Código del firmante
 * estado Estado de la etapa
 * fecha Fecha de la firma
 */
public class PtfrEtapas
```

El método incluirFirmante obtiene como parámetro un String con el firmante a añadir. Se puede llamar tantas veces como sea necesario

Método incluirFirmante

```
    /**
     * Añade una etapa a la solicitud
     * @param codigoFirmante String con el código del firmante
     */
public void incluirFirmante(String codigoFirmante)
```

El método incluirAnexo obtiene el código de archivo e indica si se ha de firmar o no

Método incluirAnexo

```
     /**
     * Añade un anexo a la solicitud
     * @param codigoAchivo código de archivo
     * @param firma indica si hay que firmar o no
     */
    public void incluirAnexo(String codigoAchivo, Boolean firma)
```

4.10. Método consultaEtapasDoc

PtfrService.java

```
    /**
     * Consulta los estados por los que ha pasado un documento
     * @param codAplicacion Código de la aplicación
     * @param referencia Referencia del documento a consultar
     * @param codigoArchivo Código del archivo
     * @return {PtfrConsultaEtapasOut}
     * @throws ServiceException
     */
    public PtfrConsultaEtapasOut consultaEtapasDoc(String codAplicacion,
            String referencia, String codigoArchivo) throws ServiceException;
```

Objeto de salida

Todos los servicios de Ptfr que devuelvan un objeto heredarán de PtfrRespuestaBase

PtfrConsultaEtapasOut.java

```
/** Respuesta del método consultaEtapasDoc
 * {atlas.eadm.services.PtfrService#consultaEtapasDoc}
 * Hereda de {PtfrRespuestaBase}
 * Lista de etapas {PtfrEtapas}
 * documento {PtfrDocumento}
 */

public class PtfrConsultaEtapasOut extends PtfrRespuestaBase {
```

PtfrDocumento.java

```
/**
 * Objeto para guardar la información de un documento
 * estado Estado del documento
 * codArchivo Código del archivo
 * codArchivoOriginal Código del archivo original
 * referencia Referencia del documento a consultar
 * referenciaSolicitante Referencia del solicitante
 * fechaAccion Fecha de la accion
 * motivorechazo Motivo del rechazo
 */
public class PtfrDocumento
```

PtfrEtapas.java

```
/**
 * Clase utilizada para guardar información de
 * una etapa
 * codigoFirmante Código del firmante
 * orden Orden de la etapa
 * firmante Código del firmante
 * estado Estado de la etapa
 * fecha Fecha de la firma
 */
public class PtfrEtapas {
```

4.11. Método enviar

PtfrService.java

```
      /**
     * Envía las notificaciones a todos los usuarios del portafirmas que lo
     * requieran
     *
     * @param codAplicacion Código de la aplicación
     * @return String con resultado
     * @throws ServiceException
     */
    public String enviar(String codAplicacion) throws ServiceException;
```

4.12. Método enviarOffline

PtfrService.java

```
        /**
     * Envía las notificaciones pendientes que existan en BBDD.
     * @return String con resultado
     * @throws ServiceException
     */
    public String enviarOffline() throws ServiceException;
```

5. ENLACES RELACIONADOS

Producto	URL
Documentación EREG	http://intranet.madrid.org/arquitecturasw/admelectronica/EREG